基于密度聚类算法的校园人群聚集和移动规律分析 *

郭玉彬，吴宇航，周哲帆，李西明†

(华南农业大学 数学与信息学院，广州 510642)

摘　要：针对某高校无线网日志数据进行挖掘分析，获取校园人群聚集点分布和人群移动规律。首先利用分布式统计算法统计校园内各建筑物的无线网络连接人次；然后建立校园建筑物的中心点经纬坐标的 R-树索引，并对 R-树叶子节点分组，以此将校园分成几个部分；再利用密度聚类算法对校园每一个部分中的建筑物中心点经纬坐标进行聚类得到校园区域划分；最后结合聚类结果和统计结果获取人群聚集区域和区域之间人群移动规律。研究结果可为学校校车路径规划、共享单车部署和校园功能区规划等工作提供参考。

关键词：无线网络；日志数据；R-树；密度聚类；人群聚集和移动

中图分类号：TP391　　doi: 10.3969/j.issn.1001-3695.2018.09.0638

Analysis of crowd aggregation and movement on campus based on density clustering algorithm

Guo Yubin, Wu Yuhang, Zhou Zhefan, Li Ximing†

(*College of Mathematics & Information, South China Agricultural University, Guangzhou* 510642, *China*)

Abstract: This paper analyzed the wireless network log data of a university to study the law of crowd aggregation and movement on campus. Firstly, the number of wireless network connections of each building was obtained by distributed statistical algorithms on campus. Then it established the R-tree index of building central latitude-longitude coordinates, and grouped the R-tree leaf nodes to divide the campus into parts. Next it clustered the building central latitude-longitude coordinates of each part to get the campus area division by the density clustering algorithm. Finally, this paper got the law of crowd aggregation area and movement between areas from clustering results and statistical results. Experimental results provide references for school bus routing, shared bicycle deployment and campus function area planning etc.

Key words: wireless network; log data; R-tree; density clustering; crowd aggregation and movement

0　引言

随着移动互联网技术的快速发展，我国多数高校已完成或正在进行无线网络全覆盖工作，与之相适应，校园网的无线上网日志数据呈爆炸式增长。笔者所在高校，宿舍与办公区已实现全覆盖，2018 年上半年仅上网认证数据就达到 252 515 722 条(2.5 多亿条)。这些日志数据主要包含用户移动设备的 MAC 地址、上下线时间、连接无线访问接入点(wireless access point，简称无线 AP)的名称和离线原因等信息。通过对日志信息的分析，可以得到校园人群分布、人群移动等规律性信息，而这些信息可以为诸如校巴调度，共享单车部署与路线设计和校园功能区规划等工作提供参考。

人群聚集和移动规律分析是当前大数据背景下的研究热点。文献[1]提出了基于 GPS 轨迹的兴趣点和出行规划挖掘方法，利用用户个人位置历史数据结合出行经验，发现了研究区域的十大兴趣点和用户在各个兴趣点之间的出行规律；文献[2]通过获取大量的手机用户位置信息，分析用户六个月的轨迹数据得出人类移动时空规律性；文献[3]分析大量的私家车 GPS 轨迹数据，给出人们频繁的出行方式，并预测近期交通热点区域和某个区域拥堵可能性。除了利用轨迹数据，还有一些研究利用 Wi-Fi 网络日志数据分析人群体聚集或移动规律，文献[4]通过 Wi-Fi 网络日志建立了一个移动模型，利用设备上下线的时间，得到两点之间的移动速度，提取用户移动特性，并准确描述用户移动行为。

在人群聚集和移动规律研究中，聚类是一种有效的分析方法。文献[5]利用聚类技术，从空间和时间两个维度对用户轨迹数据进行两阶段聚类，以此构建用户兴趣区。文献[6]利用 DBSCAN(density based spatial clustering of applications with noise)对人群聚集热点区域进行提取和分析，发现相同时间段之间热点区域存在较高的覆盖率，而不同时间段的热点区域存在较大的差异的规律。除了聚类算法外，学者们还有利用神经网络[7]和统计法[8]等方法来发现人群移动规律等。

针对校园内人群聚集和移动规律也有一些学者进行了研究。文献[9]利用 DBSCAN 算法对武汉大学学生轨迹数据进行聚类，提取聚集点，并分析学生各类活动时段分布。该论文对利用 GPS 数据研究学生行为工作具有参考意义，其不足之处在于轨迹数据来自志愿者，数据量较少且代表性不足。文献[10]利用加拿大蒙特利尔康戈迪亚大学的 Wi-Fi 网络日志数据进行分析和聚类，识别了网络用户在建筑物内活动类型，如上下课，办公，在公共区域使用网络，并且提出搜索算法，用于关联网络用户多天中同一类型的活动。此论文针对的人群移动范围较小，没有考虑建筑物之间人群移动问题。

与轨迹数据相比，校园无线网络的日志数据量大且稳定，更能反映人群聚集与移动情况。本文对某高校无线网络的日

收稿日期：2018-09-10；**修回日期：**2018-10-28　**基金项目：**国家重点研发计划资助项目（2016YFD0800307, 2016YFC0501801）；国家科技支撑计划资助项目（2015BAD06B03-3）

作者简介：郭玉彬（1973-），女，山东聊城人，副教授，博士，主要研究方向为数据库、网络计算；吴宇航（1990-），男，广东雷州人，硕士研究生，主要研究方向为大数据研究与应用；周哲帆（1996-），男，广东揭阳人，本科生；李西明（1974-），男（通信作者），山东临清人，高级工程师，博士，主要研究方向为大数据应用技术研究、机器学习（liximing@scau.edu.cn）.

志数据进行处理和分析，首先利用 MapReduce 计算模式的分布式统计算法统计每一栋建筑的中心经纬坐标[1]对应的无线网络连接设备数量，统计结果表示每一栋建筑物的所有无线 AP 的连接人次[2]的总和。然后对研究区域所有建筑物中心经纬坐标建立 R-树索引，并对 R-树叶子节点分组，粗略地将校园划分不同部分，再利用密度聚类算法对校园每一部分中心经纬坐标进行聚类得到校园详细的区域划分。最后结合聚类结果和统计结果得到人群热点分布区域和区域之间人群移动规律。本论文利用 R-树索引解决中心经纬坐标动态变化的问题，可长期、持续对校园网日志数据进行处理而不受校园网扩建和 AP 不稳定的影响，增加了实验的灵活性。另外，采用 MapReduce 计算模式实现核心算法，提高了大数据处理效率。图 1 是本文算法整体流程图。

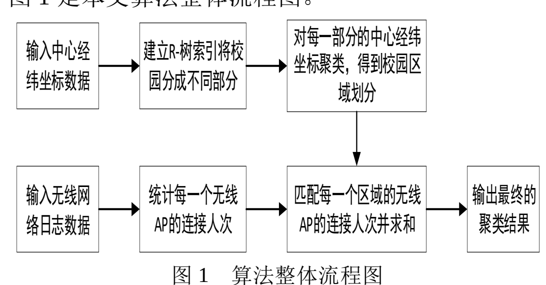


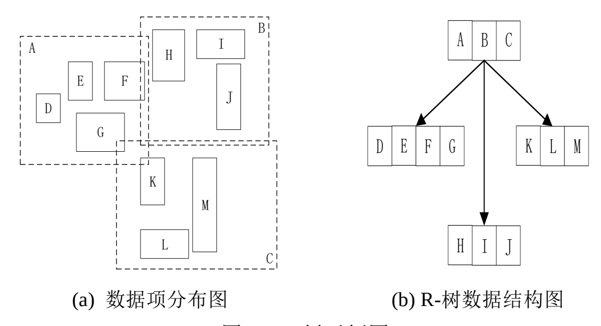
图 1　算法整体流程图

Fig. 1　Algorithm overall flow chart

1　相关技术

1.1 R-树

R-树是一个层次化的、高度平衡的多层数据结构，是 B-树在多维数据空间上的自然扩展[11]，R-树的每个节点对应一个最小外包框 (minimum bounding rectangle,MBR)，该 MBR 是包围所有子节点的最小空间范围。图 2 是 R-树的示例，其中(a)是数据项分布情况，图中实线框表示空间对象的 MBR，其可为二维坐标，如经纬坐标，表示若干二维形状的最大范围。虚线框表示中间节点索引项对应的索引空间。图 2(b)是其对应的 R-树图。



(a) 数据项分布图　　(b) R-树数据结构图

图 2　R-树示例图

Fig. 2　Example of R-tree

假设 $m(2<=m<=M/2)$为节点包含的索引项(数据项)的最小数目，R-树需要满足以下性质：

a）除根节点之外，所有中间节点包含有 m ($m=M/2$)至 M 个记录索引。

b）根节点至少有两叶子节点，除非它同时是叶子节点。

c）每一个叶子节点拥有 $m(m=M/2)$至 M 个数据项。

d）所有的节点都需要同样的存储空间。

e）所有叶子节点都位于同一层次。

[1] 中心经纬坐标：建筑物中心点的经纬坐标

[2] 连接人次：无线网络的连接人次

1.2 DBSCAN 算法

DBSCAN 算法是一种利用类的高密度连通性来快速发现任意形状类的密度聚类算法[12]。DBSCAN 算法的基本思想是对于一个类中的每个对象，在其给定半径的领域中包含的对象不能少于某一给定的最小数目。令样本集 $D=\{x_1,x_2,...,x_m\}$，样本 $x_i,x_j \in D$。取 $\varepsilon>0$，$\min Pts$ 为正整数，则有如下定义：a）x_j 的 ε-邻域，包含在 D 中且与 x_j 的距离不大于 ε 的样本集合；b）核心对象，如果 x_j 的 ε-邻域至少包含 $\min Pts$ 个样本，则 x_j 为核心对象；c）边界点，x_i 本身不是核心点,但它包含在另外一个核心点 x_j 的 ε-邻域内,则称 x_i 为边界点；d）噪声点，数据集合中既不是核心点也不是边界点的其他点称为噪声点；e）密度直达，如果 x_i 位于 x_j 的 ε-邻域中，且 x_j 是核心对象，则称 x_i 由 x_j 密度直达；f）密度可达，对于 x_i 和 x_j，如果存在样本序列 $p_1,p_2,\ldots,p_T$,满足 $p_1=x_i$，$p_T=x_j$,且对任意 $k\in[1,\ldots,T-1]$，p_{k+1} 由 p_k 密度直达，则称 x_j 由 x_i 密度可达；g）密度相连，对于 x_i 和 x_j，如果存在核心对象样本 x_k，使 x_i 和 x_j 均由 x_k 密度可达，则称 x_i 和 x_j 密度相连。基于以上定义，DBSCAN 算法步骤如下：

a)从样本集 D 中取任意样本 P，并标记 P 为已读。

b)如果 P 是核心对象，找出 P 的 ε-邻域所有密度可达点。

c)如果 P 是一个边界点，没有对象从 P 密度可达，P 被暂时标注为噪声点。

d)重复 a）~d），直到 D 中所有对象都被标记为已读。

e)针对所有核心对象的 ε-邻域的直接密度可达点找到最大密度相连对象集合。

重复 e）直到所有核心对象的 ε-邻域遍历完毕。

DBSCAN 算法优点是具有较强的抗噪声干扰和发现任意形状类等优点[13]。其缺点是需要找出所有的密度核心对象，同时参数 ε 和 $\min Pts$ 需要人为确定，存在一定的人为误差。参数 ε 和 $\min Pts$ 影响聚类质量，如果 ε 取值过大，会导致大多数点都聚到同一个类中，ε 取值过小，会导致一个类的分裂；如果 $\min Pts$ 取值过大，会导致在同一个类中点被标记为非核心对象，$\min Pts$ 取值过小，会导致发现大量的核心对象。针对 DBSCAN 算法的不足，国内外学者提出了许多改进算法，例如 M-FDBSCAN[14]算法、HDBSCAN[15]算法等。

2　实验准备

2.1 源数据说明

本文实验数据来自某高校无线网络设备日志，抽取了 2017 年 3 月份无线网络认证数据作为源数据，一共有 39 478 898 条数据。表 1 给出无线网络日志结构说明。

表 1　无线网络日志描述

Table 1　Wireless network log description

名称	类型	长度	说明
SEQ	INT	100	唯一标志符
DATE	DATE	100	日志产生日期
TIME	TIME	100	日志产生时间
MESSAGE	VARCHAR	1000	日志内容

表 1 中，SEQ 字段是日志数据的唯一标志符，DATE 表示本条日志产生的日期，TIME 是具体产生的时间，精确到微秒。MESSAGE 字段是当前日志的日志内容，里面包含了移动设备的 MAC 地址、当前连接无线 AP 名字、设备连接状态信息等。设备连接状态有上线、下线和漫游 3 种。同时，本文还对所有的无线 AP 信息进行汇总，如表 2 所示。

表 2 中，AP_NAME 字段表示无线 AP 的名称，AP_MAC 表示当前无线 AP 的 MAC 地址，也是无线 AP 的唯一标志，

CEN_GPS 表示建筑物中心点的经纬坐标，BUILDING 表示无线 AP 所在的建筑物名称。在此表中，由于同一栋建筑物中无线 AP 距离相距比较近且每一栋建筑物无线 AP 数量比较多，因此取建筑物中心点的经纬坐标代表建筑物中所有无线 AP 的实际的经纬坐标，使建筑物的无线 AP 表示更加简化同时建筑物的位置被标记。

表 2　无线 AP 表

Table 2　Wireless access point table

名称	类型	说明
AP_NAME	VARCHAR	无线 AP 名称
AP_MAC	VARCHAR	无线 AP 的 MAC 地址
CEN_GPS	VARCHAR	建筑物中心点的经纬坐标
BUILDING	VARCHAR	无线 AP 所在建筑物名称

2.2 数据预处理

数据预处理主要是选择研究时段的日志数据，并依据日志数据中 MESSAGE 字段出错标签去掉出错数据，然后从日志数据中获取并保存每一个设备位置变化信息。出错数据包括认证出错数据和无线 AP 设备出错数据等。位置变化信息例子如表 3 所示，每一条数据记录某一设备在某一个时刻的位置。具体过程如下：

a)将无线网络日志信息数据表和无线 AP 信息表保存到 HDFS 中。

b)使用正则表达式抽取无线网络日志的 MASSAGE 字段中的移动设备的 MAC 地址、移动设备连接无线 AP 所在的建筑物名称，并分别用 MAC 和 BUILDING 表示，根据建筑物名称和无线 AP 表匹配建筑物中心经纬坐标，用 CEN_GPS 表示。添加当前日志的 DATE、TIME 字段，并将以上所有信息保存到临时表。

c)将无线 AP 表和临时表以 CEN_GPS 字段进行两表相连后对 MAC 和 DATA 以及 TIME 字段进行排序。只保留 MAC 和 DATA 以及 CEN_GPS 字段都连续相同记录的第一条，得到预处理结果。

表 3 给出预处理后关于移动设备 0000.00d1.ab46 的 2017 年 3 月 31 日移动记录，表示此设备在当天 07:56:44 时刻在外语学院连接无线 AP，在 08:18:40 时刻在生命科学学院连接无线 AP，在 14:34:04 时刻在林学院连接无线 AP。由于一天中 MAC 和 DATE 以及 CEN_GPS 字段都连续相同的记录只保留其中一条，因此表 3 实际是保存设备一天中 CEN_GPS 变化的信息，对应用户一天中的位置变化情况。

表 3　移动设备认证位置变化例子

Table 3　Example of authentication location changes for mobile device

MAC	DATE	TIME	CEN_GPS	BUILDING
0000.00d1.ab46	2017.3.31	07:56:44	113.371997,23.16637	外国语学院
0000.00d1.ab46	2017.3.31	08:18:40	113.366686,23.164849	生命科学学院
0000.00d1.ab46	2017.3.31	14:34:04	113.367682,23.165515	林学院

3　实验及结果分析

本文实验使用 Hadoop 集群进行数据处理，集群包含 master 主机 1 台，slaver 机 3 台，节点配置为 CPUi7- 8700K，内存 8GB。平台使用 Centos7.0 操作系统，Hadoop2.7。

本文实验分为人次统计和密度聚类两个过程。在统计阶段，由表 2 可知，预处理后的数据同一栋建筑物所有无线 AP 的经纬坐标对应当前建筑物中心点的经纬坐标，即中心经纬坐标，那么计算每一个中心经纬坐标对应连接人次即为该建筑物中所有无线 AP 的连接人次总和。在聚类阶段首先对无线 AP 表中的中心经纬坐标建立 R-树索引，并根据学校的功能区划分将 R-树叶子节点进行分组和将组间重复数据归类到距离最近的组中避免聚类结果数据重复。最后对每一组的叶子节点进行密度聚类，利用聚类结果中每一个类的中心经纬坐标和统计结果的中心经纬坐标连接人次相匹配得到每一个类的连接人次，实验流程如图 3 所示。

图 3　实验流程图

Fig. 3　Experiment flow chart

3.1 人次统计

人次统计是统计预处理数据（表 3）的每一个中心经纬坐标对应的网络连接设备数量。由于该校的校园无线网络账户不能同时在两台设备及以上登录，且每一个中心经纬坐标代表一栋建筑物，那么统计建筑物中所有无线 AP 在对应时间内的网络连接设备数总和，实际是统计建筑物中所有无线 AP 在对应时间内的网络连接人次总和。人次统计过程见算法 1。

算法 1　按时间段选择的预处理数据

输入：预处理后的日志文件

输出：各个中心经纬坐标的连接人次总和

```
// map 阶段遍历所有数据项
map(LongWritable key, Text value, Context context):
//按照制表符对数据切割
string[] data =val.toString().split("\t");
string CEN_GPS= data[3];
//以中心经纬坐标作为 key 输出
context.write (new Text(CEN_GPS),1);
// reduce 阶段实际统计每一栋建筑物连接人次
reduce(Text key, Iterable<Text> values, Context context):   // Reduce 阶段
int num=0;
// 遍历迭代器;
for each val in values
  num++; //统计数量
end for
```

```
context.write (null, key+" "+num) ;  //输出结果
```

在算法 1 中，Map 阶段逐行遍历预处理数据，并以<key, value>形式存储，按照制表符对 value 进行切割保存在 data 数组中，并取出中心经纬坐标传给 reduce 阶段。在 reduce 阶段累加每一个 key 对应迭代器的值既是各个中心经纬坐标的连接人次。

3.2 密度聚类

密度聚类过程分成两个步骤：a）对校园内所有的中心经纬坐标建立 R-树索引，并根据学校的功能区划分将 R-树叶子节点进行分组和将组间重复数据归类到距离最近的组中；b）对每一组的数据进行密度聚类，利用聚类结果中每一个类的中心经纬坐标和统计结果的中心经纬坐标连接人次相匹配得到每一个类的中心经纬坐标的连接人次。

由于学校无线网络存在不定时扩建和网络不稳定状态，从而影响到中心经纬坐标数量，为了适应中心经纬坐标动态变化和让聚类结果更加准确，本文对中心经纬坐标建立 R 树索引，通过 R-树的增加或减少叶子节点的方法来增加或者减少中心经纬坐标。建立 R-树索引后，为了能按照学校功能区划分将同一个功能区的中心经纬坐标汇集一起，即需对 R-树叶子节点进行分组。假如将 R-树叶子节点分成 n 组，那么从 R-树某一层有 n 个父节点开始分别抽取每一个父节点对应的所有叶子节点作为一个组，并将组间重复数据归类到距离最近的组中。利用 DBSCAN 聚类对每一组数据进行聚类，详细地将校园划分不同的小区域，最后根据聚类结果和统计结果计算出每一个类的每一个中心经纬坐标对应连接人次。

本文建立 R-树索引及分组的过程如下：

a)从无线 AP 表中获取所有中心经纬坐标。

b)对中心经纬坐标建立 R-树索引。

c)将 R-树的叶子节点分成 n 组。

d)如叶子节点同时归属两个组及两组以上，将改节点归类到其距离最近的组。

基于 R-树的密度聚类算法，见算法 2，作用是先建立中心经纬坐标的 R-树索引，并将 R-树叶子节点进行分组，然后用 DBSCAN 聚类算法对每一组叶子节点进行密度聚类。

算法 2　基于 R-树的密度聚类

```
输入:中心经纬坐标集 dataTable。
输出:DBSCAN 聚类结果。
//Node 类是一种构造 R-树的数据结构
class Node {
   final double[] coords; //当前节点值
   final boolean leaf; //是否为叶子节点
   final LinkedList<Node> children;//子节点
   Node parent;//父节点
}
Node R_tree;
List<String> cluster，clustersList =null;
List<List<String>>clusters，dbClusters =null;
//建立 R-树
RTree RTree =new RTree();
DBSCAN DBSCAN=new DBSCAN();
for each data in dataTable
// 添加数据构建 R-树
R_tree =RTree.insert(data);
end for
// 将 R-树索引中叶子节点分成 n 个组
int n=7; //组个数，暂取值为 7
// clusters 存储每一组中心经纬坐标
clusters =RTree.createClusterList(n);
for each cluster in clusters
// 对每一组中心经纬坐标做 DBSCAN 算法
clustersList =DBSCAN.execute(cluster)；
dbClusters.add(clustersList);// 保存聚类结果
end for
return dbClusters；
```

在算法 2 中，Node 类是一种构造 R-树的数据结构，其包含了当前节点值，子节点和父节点等属性。RTree 类中 insert()方法代表将输入的数据进行构造 R-树数据结构并保存在一个全局变量中，createClusterList()方法代表将 R-树的所有叶子节点分成 n 组，并返回每一组的数据。DBSCAN 类中 execute()方法是对每一组 R-树的叶子节点进行聚类，返回聚类结果。算法 2 步骤如下：首先调用 RTree.insert()方法建立关于中心经纬坐标的 R-树，然后调用 RTree. createClusterList()方法时将 R-树叶子节点分成 n 组，如叶子节点同时归属两个组及以上，不包含该叶子节点计算其所在所有组的中心，并将其归类到距离最近的中心所对应的组中。最后调用 DBSCAN.execute()方法对每一组中心经纬坐标做 DBSCAN 聚类，返回并保存聚类结果。

对每一组数据进行聚类之后，最后需要计算每一个类的总连接人次，此过程见算法 3。

算法 3　类的连接人次匹配

```
输入:算法 1 统计结果，算法 2 聚类结果。
输出: 类的连接人次匹配。
MapA(LongWritable key, Text value, Context context): // MapA 阶段，输出聚类结果
String[] strA =val.toString().split("\t")
String AP = strA [0];//存储无线 AP
//存储无线 AP 对应的类
String clusterNum= strA [1];
context.write(new Text(AP), new Text(clusterNum));
MapB(LongWritable key, Text value, Context context): // MapB 阶段，输出连接人次统计结果
String[] strB =value.toString().split("\t");
String AP = strB [0]; //存储无线 AP
context.write(new Text(AP), value);
Reduce(Text key, Iterable<Text> values, Context context): // Reduce 阶段
for each val in values
String clusterNum;//类标记
Text statisticData=new Text();
//判断是否为聚类结果数据
If val.indexof("\t")!=-1
statisticData=val；  // 保存统计结果
else
clusterNum=val；  // 标记所属类
end if
end for
context.write(null, new Text(statisticData +" "+ clusterNum))；
//输出结果
```

在算法 3 中，MapA 和 MapB 阶段分别遍历 DBSCAN 聚类结果和中心经纬坐标的连接人次统计结果，并将其输出给 Reduce 阶段。在 Reduce 阶段，首先根据对象判断数据是来自 DBSCAN 聚类结果还是统计结果，即如果对象包含一个

制表符，则用 statisticData 保存统计结果，否则用 clusterNum 保存 DBSCAN 聚类结果，statisticData 和 clusterNum 出现在同一个 Reduce 对象中，说明它们的中心经纬坐标是相同的，所以连接字段已经匹配成功。最后设置 Reduce 阶段的 key 为 null，value 为 statisticData +" "+clusterNum，即可输出每一个中心经纬坐标和其对应的类的连接人次结果。求和即可得到对应类的总连接人次。

3.3 实验结果及分析

本文源数据总有 39 478 898 条，预处理后有 13 290 271 条，学校的无线 AP 总数为 10,255 个，提取得到的中心经纬坐有 125 个。先对中心经纬坐标建立 R-树索引，由于学校功能区分成 7 大块，根据学校的功能区划分将 R-树叶子节点分成 7 组，用 ABCDEFG 表示。对于 DBSCAN 密度聚类算法，本文 ε 和 $\min Pts$ 值参考了文献[16]的思路，分别取 863.92 和 3。密度聚类将研究区域划分为 10 个类，将每组和每一组分类的分布情况映射到地图上，与实际情况相符合，如图 4 所示。

表 4 给出聚类结果以及不同时间段内每一个类和统计结果相匹配得到连接人次匹配结果。可以看出 A、B、F、G 和 H 组聚类后分别得到一个类，记作 A1、B1、F1、G1 和 H1。C、D、E 三组组聚类后都得到两个类，分别标记为 C1、 C2；D1 、D2； E1、E2。统计时间段连接人次分一个月总人次、日均人次、日均不同时间段内人次（上午、中午、下午、晚上和凌晨人次）。

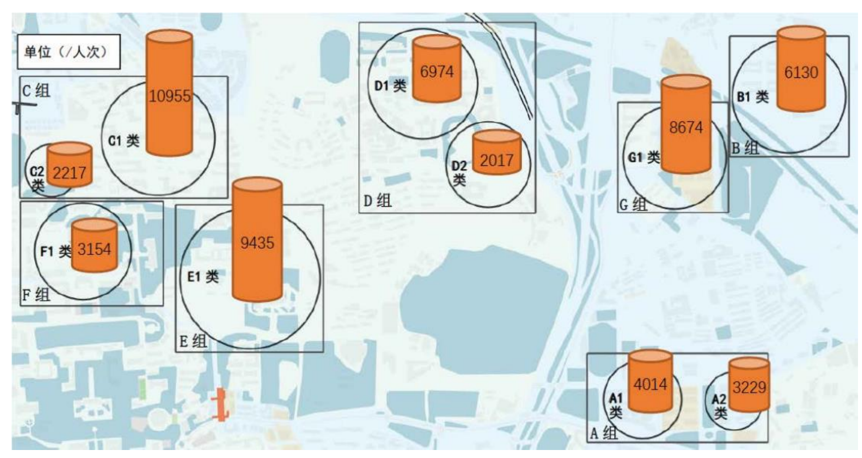


图 4 分组结果和聚类结果图

Fig. 4 Result of grouping and clustering

表 4 不同时间段内聚类结果及统计结果的人次匹配 /人次

Table 4 Matching of clustering results and statistical results in different time periods /person time

R-树分组	聚类结果	一个月	日均	上午(6:00-12:00)	中午(12:00-14:00)	下午(14:00-18:00)	晚上(18:00-24:00)	凌晨(00:00-6:00)
A	A1	120431	4014	1424	453	997	1129	12
	A2	96859	3229	903	371	479	1016	459
B	B1	183904	6130	1779	687	912	1676	1076
C	C1	328661	10955	3498	1442	2311	2976	729
	C2	66509	2217	623	242	303	631	419
D	D1	209216	6974	1802	641	925	1839	1768
	D2	60519	2017	737	220	492	506	63
E	E1	283043	9435	3341	1071	2619	2203	202
F	F1	94609	3154	999	365	667	853	270
G	G1	260234	8674	2596	850	1763	2484	981

从表 4 可以看出日均人次分布中，C1、E1、G1、D1 和 B1 区日均人次较多，说明这 5 个区人群分布比较密集，属于人群聚集区。上午日均人次 C1、E1、G1 和 D1 比较多，由此可知上午这 4 个区上午是人群聚集区；同样，中午人群聚集区是 C1、E1、G1 和 D1；下午人群密集地是 E1、C1 和 G1；晚上人群聚集区是 C1、G1、E1、D1 和 B1；凌晨人群聚集区是 D1、B1 和 G1。

为了研究每一区域的人群移动情况，本文对每一个区的人群移动次数进行统计，统计结果如表 5 所示，人群移动次数比较频繁的区域拓扑图如图 5 所示。

由表 5 和图 5 可以看出，E1 区、A1 区、B1 区、C1 区、D1 区和 G1 区人群移动比较频繁，可以看出 E1 区在此校是人群移动最频繁区域。E1 区人群移动人次也是最多的，其中流入人次是 997 次，流出人次是 1047 次；C1 区人群移动人次仅次于 E1 区，其中流入人次 926 次，流出人次是 1032 次，其他相对人群移动人次较多的区还有 D1、G1 和 A1 区。在不同区域间的人群移动，E1 和 C1 之间人群移动基数最大，除此之外，A1 和 A2、C1 和 D1、B1 和 G1 的区域间的人群移动基数较大。对于区域内人群移动差距比较大的区域有 D1 区，相差 132 次和 C1 区，相差 106 次。

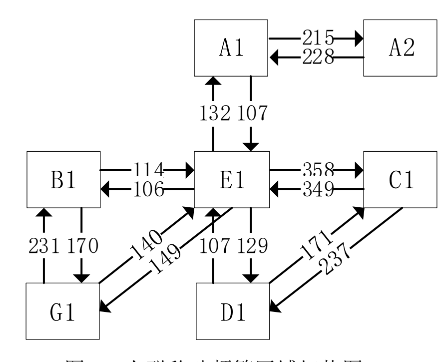


图 5 人群移动频繁区域拓扑图

Fig. 5 Area topology for crowd frequent movement

表 5　跨区人群移动统计 /人次

Table 5　Cross areas mobility population statistics　/person time

	A1	A2	B1	C1	C2	D1	D2	E1	F1	G1	总和
A1	0	215	19	55	3	15	21	107	18	59	512
A2	228	0	4	18	1	10	15	38	7	21	342
B1	14	3	0	56	1	7	7	106	4	170	368
C1	67	17	66	0	83	237	31	349	96	86	1032
C2	3	1	1	65	0	3	1	46	29	3	152
D1	14	7	9	171	2	0	80	107	34	44	468
D2	20	12	8	28	0	111	0	31	6	74	290
E1	132	34	114	358	48	129	35	0	75	149	1074
F1	19	5	5	97	26	48	6	73	0	12	291
G1	65	21	231	78	3	47	73	140	14	0	672
总和	562	315	457	926	167	607	269	997	283	618	

表 6　上午跨区人群移动情况 /人次

Table 6　Cross areas mobility population in the morning /person time

	A1	A2	B1	C1	C2	D1	D2	E1	F1	G1	总和
A1	0	135	14	26	1	7	10	46	10	26	275
A2	77	0	2	5	0	3	7	12	3	6	115
B1	8	1	0	28	0	5	2	51	2	78	175
C1	39	12	51	0	59	146	21	191	78	60	657
C2	2	1	1	42	0	3	0	18	19	1	87
D1	9	7	5	91	1	0	48	58	18	37	274
D2	9	11	3	18	0	72	0	19	3	38	173
E1	69	27	86	229	33	83	23	0	54	103	707
F1	12	3	4	68	16	25	3	41	0	8	180
G1	28	11	129	40	2	29	28	71	6	0	344
总和	253	208	295	547	112	373	142	507	193	357	

为了研究上下午该校各区的人群移动情况，本文也统计了该校上下午人群移动人次，如表 6 和 7 所示，其中表 6 表示上午跨区人群的移动情况，表 7 表示下午跨区人群的移动情况。

从表 6 可以看出，C1 区的上午人群移动人次最多，流入人次是 926 次，流出时 657 次，结合表 4 可以计算出该区域内部移动人次，即上午日均人群移动次数减去该区域上午人群流入次数再减去区域上午人群流出次数，那么 C1 区的区域内部人群移动次数为 2294，占比是 65.58%。同样，E1 区上午人群移动人次仅次于 C1 区，流入人次为 507，流出人次为 707 次，区域内部人群移动次数为 2127 次，占比 63.66%。除了 C1 区与 E1 区之外，还有 D1 和 G1 区上午人群移动次数比较大。从表 6 中也看出上午人群流入流出的次数相差比较大的区域有 E1，相差 200 人次，其次是 B1 和 C1 区，相差分别是 120 次和 110 次。从表 7 可以看出，C1 区和 E1 区依然是下午人群移动人次最多的两个区域，结合表 4，也可以计算出两个区域人群移动次数分别是 1 387 次和 1 638 次，占比分别是 60.01%和 62.54%。除了这两个区域，还有 A1 和 G1 区下午人群移动次数比较大。从表 7 中可以看出各区下午群流入流出的次数相差不大。

从以上分析中，可以看出该校的 C1 区、E1 区、D1 区和 B1 区日均网络连接人次较多，其中 C1 和 B1 区是宿舍区，E1 是学校的中心区，区内有图书馆、行政楼和教学楼等，D1 区是宿舍区和教学楼的组合。由此反映出学校的人群分布主要是宿舍区和中心区，与实际情况相符，在人群聚集区可以增加娱乐设施、商铺，扩大消防通道和增强安保力度等。对于人群移动情况，可以从分析中获知 E1 区、A1 区、B1 区、C1 区和 D 区人群移动比较频繁，其中上午人群移动比较频繁区域有 C1 区、E1 区和 D1 区，下午人群移动比较频繁区域有 C1 区、E1 区和 A1 区。可以根据实际情况在人群移动次数多的地方增加交通安全提醒，可以在人群移动频繁地方多投放共享单车或者修建或者增加共享单车停放点。对于该校的人流的方向性，可以按照 A1<=>A2<=>E1<=>D1，G1<=>B1<=>E1<=>C1 的区域道路连通情况设计校巴路线或者共享单车车道，对于上下午人群流量次数较多的区域可以考虑增加校巴趟次或者调整校巴调度次数。

表 7　下午跨区人群移动情况　/人次

Table 7　Cross areas mobility population in afternoon　/person time

	A1	A2	B1	C1	C2	D1	D2	E1	F1	G1	总和
A1	0	77	11	26	1	8	8	43	7	33	214
A2	76	0	2	8	0	4	4	12	2	10	118
B1	9	2	0	29	0	4	3	48	2	85	182
C1	30	8	30	0	30	74	18	171	62	47	470
C2	1	0	1	37	0	1	0	19	15	1	75
D1	8	3	4	64	1	0	33	46	14	23	196
D2	9	5	2	14	0	36	0	15	2	23	106
E1	51	16	54	178	17	46	16	0	45	82	505
F1	8	2	2	55	11	12	3	41	0	6	140
G1	33	10	89	43	1	23	23	81	6	0	309
总和	225	123	195	454	61	208	108	476	155	310	

4　结束语

本文通过对某高校无线网络的日志数据进行统计和聚类分析，得到该校的区域划分情况和人群聚集和人群移动情况，为非商业化或者商业化活动策划作参考。本文首先去除一些与实验无关数据，并从日志文件提取设备位置变化信息；然后利用设备位置变化信息统计每一栋建筑物中心经纬坐标对应的连接人次。再建立校园内所有中心经纬坐标的 R-树索引，并对 R-树叶子节点分组后采用密度聚类算法对每一组数据聚类得到校园区域划分情况。最后结合聚类结果和统计结果，得到每一个类的中心经纬坐标的连接人次，并对结果进一步求出类之间的人群移动人次。通过对结果分析，所得结论对校车路径规划、共享单车部署、校园功能区规划等提供参考。另外，利用 R 树可提高数据处理效率，并且在 R-树的中可以动态添加或者减少无线 AP 从而增加了实验的灵活性。本文下一步的工作是跟踪人员移动情况，结合成绩数据和上网流量分析学生的在校行为。

参考文献：

[1] Zheng Y, Zhang L, Xie X, *et al.* Mining interesting locations and travel sequences from GPS trajectories [C]// Proc of the 18th International Conference on World Wide Web. New York:ACM Press, 2009: 791-800.

[2] González M C, Hidalgo C A, Barabási A. Understanding individual human mobility patterns [J]. Nature, 2008, 453 (7196): 779-782.

[3] Giannotti F, Nanni M, Pedreschi D, *et al.* Unveiling the complexity of human mobility by querying and mining massive trajectory data [J]. VLDB Journal, 2011, 20(5): 695.

[4] Kim M, Kotz D, Kim S. Extracting a mobility model from real user traces [C]// Proc of IEEE International Conference on Computer Communications. Piscataway,NJ: IEEE Press, 2006: 1-13.

[5] 冀亚丽，桂小林，戴慧珺，等. 支持轨迹隐私保护的两阶段用户兴趣区构建方法 [J]. 计算机学报, 2017, 40(12): 2734-2747. (Ji Yali, Gui Xiaolin, Dai Huijun, *et al.* Constructing user's interest regions with two steps for trajectory privacy protection [J]. Chinese Journal of

Computers, 2017, 40(12): 2734-2747.)

[6] 张文星.人群聚集热点区域分析与预测[D].银川：宁夏大学，2017. (Zhang Wenxing. Analyze and Predict the hot spot regions [D]. Yinchuan: Ningxia University, 2017.)

[7] Liao L, Patterson D J, Fox D, *et al*. Building personal maps from GPS data [J]. Annals of the New York Academy of Sciences, 2006, 1093(1): 249.

[8] 齐佳倩. 基于视频监控数据的人群行为分析和异常轨迹检测 [D]. 北京：北京交通大学, 2018. (Qi Jiaqian. Crowd behaviors analysis and and abnormal trajectory detection detection based on surveillance data [D]. Beijing: Beijing Jiaotong University, 2018.)

[9] 杜胜兰，李枫，黄长青，等. 基于轨迹数据的武汉大学学生行为规律分析 [J]. 测绘地理信息, 2017, 42(1): 91-95. (Du Shenglan, Li Feng, Huang Changqing, *et al.* Trajectory-based activity pattern analysis of Wuhan University's students [J]. Journal of Geomatics, 2017, 42 (1): 91-95.)

[10] Poucin G, Farooq B, Patterson Z. Activity patterns mining in Wi-Fi access point logs [J]. Computers Environment & Urban Systems, 2018, 67: 55-67.

[11] Guttman A. R-trees: a dynamic index structure for spatial searching [C]// Proc of ACM SIGMOD International Conference on Management of Data. New York: ACM , 1984: 47-57.

[12] 张文元，谈国新，朱相舟. 停留点空间聚类在景区热点分析中的应用[J]. 计算机工程与应用，2018，54(4): 263-270. (Zhang Wenyuan, Tan Guoxin, Zhu Xiangzhou. Application of stay points spatial clustering in hot scenic spots analysis [J]. Computer Engineering & Applications, 2018, 54(4): 263-270.)

[13] 刘淑芬，孟冬雪，王晓燕. 基于网格单元的 DBSCAN 算法 [J]. 吉林大学学报:工学版，2014，44(4): 1135-1139. (Liu Shufen, Meng Dongxue, Wang Xiaoyan. DBSCAN algorithm based on grid cell [J]. Journal of Jilin University: Engineering and Technology Edition, 2014, 44(4): 1135-1139.)

[14] Erdem A, Gündem T I. M-FDBSCAN: a multicore density-based uncertain data clustering algorithm [J]. Turkish Journal of Electrical Engineering & Computer Sciences, 2014, 22(1): 143-154.

[15] Melvin R L, Xiao J, Godwin R C, *et al.* Visualizing correlated motion with HDBSCAN clustering [J]. Protein Science, 2018, 27(1): 62-75.

[16] 赖丽萍，聂瑞华，汪疆平，等.基于 MapReduce 的改进 DBSCAN 算法 [J]. 计算机科学，2015，42(S2): 396-399. (Lai Liping, Nie Ruihua, Wang Jiangping, *et al*. Improved DBSCAN algorithm based on MapReduce [J]. Computer Science, 2015, 42 (S2): 396-399.)